Improving Distributed Diagnosis Through Structural Model Decomposition

Anibal Bregon [1], **Matthew Daigle** [2], **Indranil Roychoudhury** [3]
Gautam Biswas [4], **Xenofon Koutsoukos** [4], **and Belarmino Pulido** [1]

[1] *Department of Computer Science, University of Valladolid, Valladolid, 47011, Spain*
{*anibal,belar*}@*infor.uva.es*
[2] *University of California, Santa Cruz, NASA Ames Research Center, Moffett Field, CA, 94035, USA*
matthew.j.daigle@nasa.gov
[3] *SGT Inc., NASA Ames Research Center, Moffett Field, CA, 94035, USA*
indranil.roychoudhury@nasa.gov
[4] *Institute for Software Integrated Systems, Dept. of EECS, Vanderbilt University, Nashville, TN, 37235, USA*
{*gautam.biswas,xenofon.koutsoukos*}@*vanderbilt.edu*

ABSTRACT

Complex engineering systems require efficient fault diagnosis methodologies, but centralized approaches do not scale well, and this motivates the development of distributed solutions. This work presents an event-based approach for distributed diagnosis of abrupt parametric faults in continuous systems, by using the structural model decomposition capabilities provided by Possible Conflicts. We develop a distributed diagnosis algorithm that uses residuals, computed by extending Possible Conflicts, to build local event-based diagnosers based on global diagnosability analysis that generate globally correct local diagnosis results. The proposed approach is applied to a multi-tank system, and results demonstrate an improvement in the design of local diagnosers. Since local diagnosers use only a subset of the residuals, and use subsystem models to compute residuals (instead of the global system model), the local diagnosers are more efficient than previously developed distributed approaches.

1 INTRODUCTION

The need for increased performance, safety, and reliability of complex engineering systems motivates the development of efficient fault diagnosis methodologies. Accurate and timely centralized fault diagnosis of complex systems is difficult and can be computationally expensive. Typically, centralized solutions have been proposed for fault diagnosis, but these solutions do not scale well as the size of the system increases. These shortcomings, together with the widespread use of distributed, networked components, encourages the development of distributed diagnosis frameworks.

In previous work, we have developed a distributed design approach based on global diagnosability analysis (Roychoudhury *et al.*, 2009), where the local diagnosers are designed to provide globally correct local diagnosis results, without a centralized coordinator, and by communicating a minimal number of measurements among themselves. Later on, this work was integrated into the formal event-based framework developed in (Daigle *et al.*, 2009) to include measurement orderings within the local diagnosers. Inclusion of measurement orderings improves diagnosability, allowing the local diagnosers to be more efficient (Daigle *et al.*, 2010). However, the approach proposed in (Daigle *et al.*, 2010) still uses residual generators based on a global model of the system.

On the other hand, system decomposition methods, such as *Possible Conflicts* (PCs) (Pulido and Alonso-González, 2004), have been proposed to decompose a system model into minimal over-determined subsystems that suffice for fault diagnosis. PCs capture a subset of constraints or relations among the system variables that produce inconsistencies when faults occur. More formally, PCs are minimal subsets of equations containing sufficient analytical redundancy to generate fault hypotheses from observed measurement deviations. However, PCs require the use of a central coordinator to compute the set of minimal diagnosis candidates based on triggered PCs.

In this work, we build on ideas from system decomposition with Possible Conflicts (Pulido and Alonso-González, 2004) and event-based distributed diagnoser design as in (Daigle *et al.*, 2010) to improve the design of independent local event-based diagnosers. This work contributes by incorporating PCs into the event-based distributed diagnosis framework, leading to more robust local diagnosers (if one local diagnoser fails, it does not affect the others), better design (obtaining smaller local event-based diagnosers, that are also independent on every level, even residual generation), and a generalization of PCs to multi-output residual generators. Results, using a multi-tank system as a case study, demonstrate the improved design of the proposed approach.

The paper is organized as follows. Section 2 describes the system modeling methodology and introduces the case study. Then, Section 3 presents the theoretical concepts of the residual design approach, and Section 4 describes the theoretical background for qualitative fault isolation and event-based diagnosis. Later, in Section 5, we propose the local diagnoser design approach, and in Section 6 we demonstrate the approach for different scenarios of the case study. Finally, Section 7 concludes the paper.

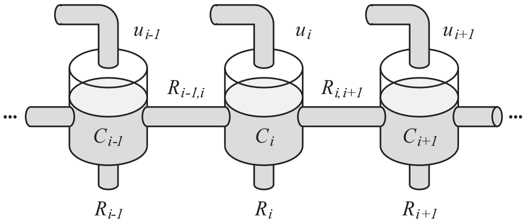


Figure 1: Tank system schematic.

2 SYSTEM MODELING

We consider the problem of single fault diagnosis in continuous systems. We assume the system, $\mathcal{S}$, is described by

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) + \mathbf{v}(t)$$
$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) + \mathbf{w}(t),$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ is the state vector, $\boldsymbol{\theta}(t) \in \mathbb{R}^{n_\theta}$ is the parameter vector, $\mathbf{u}(t) \in \mathbb{R}^{n_u}$ is the input vector, $\mathbf{v}(t) \in \mathbb{R}^{n_v}$ is the process noise vector, assumed to be zero-mean Gaussian, $\mathbf{f}$ and $\mathbf{h}$ represent the set of state and output equations, repectively, $\mathbf{y}(t) \in \mathbb{R}^{n_y}$ is the output vector, and $\mathbf{w}(t) \in \mathbb{R}^{n_w}$ is the measurement noise vector, assumed to be zero-mean Gaussian. The dimension of a vector $\mathbf{a}$ is denoted by n_a.

We denote a measurement as m, which is a time-varying signal of $\mathbf{y}(t)$ obtained from an associated sensor. The measurement set is denoted as M.

We consider single, abrupt, parametric faults, where faults are modeled as unexpected step changes in system parameter values. We name faults by the associated parameter and the direction of change, i.e., θ^+ (resp. θ^-) denotes a fault defined as an abrupt increase (resp. decrease) in the value of parameter θ. We denote a fault as f and a set of faults as F.

Throughout the paper, we will use a multi-tank system as a running example. The tanks are connected serially as shown in Fig. 1, and we will consider a variable number of tanks. For tank i, u_i denotes the input flow, C_i denotes the capacitance, and R_i denotes the resistance of the connected drain pipe. For tanks i and j, R_{ij} denotes the resistance of the connecting pipe. For an n-tank system, the pressure of tank i is described by

$$\dot{p}_i = \frac{1}{C_i}\Big(u_i + q_{i-1,i} - q_i - q_{i,i+1}\Big).$$

For tank 0, $q_{0,1} = 0$, and for tank n, $q_{n,n+1} = 0$. The output flow of tank i is:

$$q_i = \frac{1}{R_i}(p_i).$$

The flow between tanks i and $i+1$ is:

$$q_{i,i+1} = \frac{1}{R_{i,i+1}}(p_i - p_{i+1}).$$

The complete fault set F consists of $\{C_i^-, C_i^+, R_i^-, R_i^+ : i = 1, \ldots, n\} \cup \{R_{i,i+1}^-, R_{i,i+1}^+ : i = 1, \ldots, n-1\}$. The complete measurement set M is defined as $\{p_i, q_i : i = 1, \ldots, n\} \cup \{q_{i,i+1} : i = 1, \ldots, n-1\}$.

3 RESIDUAL DESIGN

We denote a residual r as a signal (generated by using the inputs and measurements of the system) that is zero when the system is fault-free, and non-zero when a fault appears in the system. The residual set is denoted as R. In previous works, we have developed a diagnosis framework, called TRANSCEND, where an observer, based on the global model of the system, is used to estimate the behavior of the system based on the set of measurements (Mosterman and Biswas, 1999). This estimation is then used to compute a residual for the measurement. In TRANSCEND, a residual r is computed as the difference between an observation, y, and the predicted nominal behavior of the output, $\hat{y}$.

Recently, system decomposition methods, like PCs, have been proposed to decompose a system model into minimal over-determined subsystems sufficient for fault diagnosis (Pulido and Alonso-González, 2004). These approaches decompose the global model into several independent minimal submodels, each with a single output. Each one of these minimal submodels estimates one measured variable, $\hat{y}$, that is compared with the observation, y, to build the residual r. Observers based on PCs are independent of each other, unlike a distributed observer scheme that uses the global model.

In both approaches, we define residuals with respect to a particular measurement. The main difference is the observer that produces the estimation $\hat{y}$. With TRANSCEND, it is computed using the global model, whereas with PCs, it is computed using a minimal observer that estimates only a single variable using other measurements as additional input.

These two approaches represent two endpoints in the space of residual design. In this section, we first describe the fundamentals of the PC approach, then we generalize PCs to submodels with multiple outputs. We show in Section 5 how this generalization is necessary for efficient diagnoser design.

3.1 Possible Conflicts

PCs are minimal subsets of equations with sufficient analytical redundancy to generate fault hypotheses from observed measurement deviations. However, the PC approach requires the use of a central coordinator to reason over the residual deviations among the different PCs to provide diagnosis results.

In this work, the global model of the system is denoted as $\mathcal{M}$, and minimal submodels obtained from PCs are denoted as $\mathcal{M}_i = (X_i, U_i, Y_i)$, where X_i, U_i, and Y_i, are the state, input, and output variables of the submodel with measured variable i as output, respectively. Using the PC approach with a three-tank system with $M = \{p_1, p_2, p_3\}$ we find a set of three minimal submodels: $\mathcal{M}_{p_1} = (\{p_1\}, \{u_1, p_2\}, \{p_1\})$, $\mathcal{M}_{p_2} = (\{p_2\}, \{u_2, p_1, p_3\}, \{p_2\})$, and $\mathcal{M}_{p_3} = (\{p_3\}, \{u_3, p_2\}, \{p_3\})$. For example, since the pressure in tank 1, p_1, is measured, a PC that estimates the pressure in tank 1 (that corresponds to minimal submodel $\mathcal{M}_{p_1}$) is defined as follows:

$$\dot{p}_1 = \frac{1}{C_1}\Big(u_1 - \frac{1}{R_1}(p_1) - \frac{1}{R_{12}}(p_1 - p_2)\Big),$$

where p_1 is the state variable, u_1 is the input to the tank, p_2 is the measured pressure of tank 2 that is it used as input for the PC, and $\{C_1, R_1, R_{12}\}$ is the subset of parameters that affects the estimation of this PC. Note that this PC is independent from p_3.

3.2 Generalizing Possible Conflicts

With PCs, each submodel is minimal, in the sense that it contains the minimum number of equations to compute only a single output. Therefore, one PC is derived for each system measurement. However, it is also possible to derive minimal multi-output submodels. These may be constructed by merging the minimal submodels in various combinations. Additional residuals may then be defined for measurements within these minimal multi-output submodels. By merging all minimal submodels, we regain the global model.

Formally, the merge operation $\oplus$ between two submodels is defined as follows.

Definition 1 (Submodel Merging)**.** Given two submodels $\mathcal{M}_i = (X_i, U_i, Y_i)$ and $\mathcal{M}_j = (X_j, U_j, Y_j)$, the merged submodel $\mathcal{M}_{i,j} = \mathcal{M}_i \oplus \mathcal{M}_j$ is defined as $\mathcal{M}_{i,j} = (X_{i,j}, U_{i,j}, Y_{i,j})$, where $X_{i,j} = X_i \cup X_j$, $U_{i,j} = (U_i \cup U_j) - (X_i \cup X_j)$, and $Y_{i,j} = Y_i \cup Y_j$.

The submodel must have all the states and outputs of its constituent submodels, and must have all the inputs, minus those that have become states in the merged submodel. We denote merged submodels by the outputs of their constituent submodels, e.g., the submodel formed by merging minimal submodels $\mathcal{M}_{p_1}$ and $\mathcal{M}_{p_2}$ is denoted as $\mathcal{M}_{p_1,p_2}$. For the global model, we drop the subscripts and denote it as $\mathcal{M}$.

For the three-tank system, where the pressures are measured, the complete set of submodels is:

$$\begin{aligned}
\mathcal{M}_{p_1} &= (\{p_1\}, \{u_1, p_2\}, \{p_1\})\\
\mathcal{M}_{p_2} &= (\{p_2\}, \{u_2, p_1, p_3\}, \{p_2\})\\
\mathcal{M}_{p_3} &= (\{p_3\}, \{u_3, p_2\}, \{p_3\})\\
\mathcal{M}_{p_1,p_2} &= (\{p_1, p_2\}, \{u_1, u_2, p_3\}, \{p_1, p_2\})\\
\mathcal{M}_{p_1,p_3} &= (\{p_1, p_3\}, \{u_1, u_3, p_2\}, \{p_1, p_3\})\\
\mathcal{M}_{p_2,p_3} &= (\{p_2, p_3\}, \{u_2, u_3, p_1\}, \{p_2, p_3\})\\
\mathcal{M} &= (\{p_1, p_2, p_3\}, \{u_1, u_2, u_3\}, \{p_1, p_2, p_3\}).
\end{aligned}$$

A residual may be defined for each measurement in each submodel. We denote a residual as $r_{m(M_i)}$, where m is the measured variable estimated by the residual, and M_i refers to the submodel with those measurements as outputs used to compute the residual. For example, $r_{p_1(p_1,p_2)}$ denotes the residual that estimates the measured variable p_1 from submodel $\mathcal{M}_{p_1,p_2}$. When the global system model is used, we drop the submodel subscript, e.g., r_{p_1} denotes the residual that estimates the measured variable p_1 and uses the global model $\mathcal{M}$.

4 QUALITATIVE EVENT-BASED DIAGNOSIS

Residuals, as described in the previous section, are triggered when faults occur in the system. Faults manifest as persistent abrupt changes in the values of the system parameters. The effects of the faults cause deviations in the observed measured variables from the nominal values. This section recapitulates the basic theoretical concepts needed to describe our diagnosis approach. We first review the theoretical framework for qualitative fault isolation and then the framework for event-based fault modeling.

4.1 Qualitative Fault Isolation

Residual deviations caused by faults are abstracted using qualitative +, −, and 0 values to form *fault signatures* (Mosterman and Biswas, 1999). Fault signatures represent these deviations as the immediate change in magnitude and the first nonzero derivative change.

Definition 2 (Fault Signature)**.** A *fault signature* for a fault f and residual r is the qualitative change in magnitude and slope of r caused by the occurrence of f, and is denoted by $\sigma_{f,r} \in \Sigma_{f,r}$.

In general, ambiguities may exist in the fault signatures, so $\sigma_{f,r}$ may not be unique. A fault signature is written as $s_1 s_2$, where s_1 is the qualitative magnitude change and s_2 is the qualitative slope change, e.g., $+-$.

We also capture the temporal order of residual deviations for a given submodel, termed *relative measurement orderings* (Daigle, 2008). Relative measurement orderings are based on the intuition that fault effects will manifest in some parts of the system before others. As described in Section 3, for a given submodel there is a residual defined for each measurement in the submodel. Within this submodel, the relative ordering of the residual deviations for these measurements can be computed based on analysis of the transfer functions from faults to residuals defined for measurements.

Definition 3 (Relative Measurement Ordering)**.** If fault f manifests in residual r_i before residual r_j, then we define a *relative measurement ordering* between r_i and r_j for fault f, denoted by $r_i \prec_f r_j$. We denote the set of all measurement orderings for f as $\Omega_{f,R}$.

Because orderings may be defined only within a given submodel, we cannot define any orderings between residuals of two different submodels because they are decoupled. For example, we cannot derive an ordering between $r_{p_1(p_1)}$ and $r_{p_2(p_2)}$ for R_{12}^+.

The fault signatures and measurement orderings can be computed automatically from a system model (Daigle, 2008). Table 1 shows the fault signatures and measurement orderings for the global model of a three-tank system with $F = \{C_1^-, C_2^-, C_3^-, R_1^+, R_2^+, R_3^+, R_{12}^+, R_{23}^+\}$, $M = \{p_1, p_2, p_3\}$, and $R = \{r_{p_1}, r_{p_2}, r_{p_3}\}$. The fault signatures derived from the minimal submodels with $R = \{r_{p_1(p_1)}, r_{p_2(p_2)}, r_{p_3(p_3)}\}$ are shown in Table 2. In this case, the PCs are able to decouple the system, and so each residual is only affected by a subset of the faults. For example, a decrease in the capacitance of tank 1, denoted by C_1^-, causes a discontinuous increase in the residuals related to tank 1 pressure, r_{p_1} and $r_{p_1(p_1)}$, followed by a smooth decrease, denoted by the signature $+-$. This is followed by smooth increases in residuals r_{p_2} and r_{p_3}, but no effect appears in residuals $r_{p_2(p_2)}$ and $r_{p_3(p_3)}$. Note that since the minimal submodels have only a single output measurement each, there are no orderings to be computed.

4.2 Event-based Fault Modeling

Fault signatures combined with relative measurement orderings provide event-based information for diagnosis. For a given fault, the combination of all fault signatures and measurement orderings yields all the possible ways a fault can manifest in the residuals. We define each of these possibilities as a *fault trace*.

Table 1: Fault Signatures and Relative Measurement Orderings for the Global Model of the Tank System.

Fault	r_{p_1}	r_{p_2}	r_{p_3}	Measurement Orderings
C_1^-	+−	0+	0+	$r_{p_1} \prec r_{p_2}, r_{p_1} \prec r_{p_3}, r_{p_2} \prec r_{p_3}$
R_1^+	0+	0+	0+	$r_{p_1} \prec r_{p_2}, r_{p_1} \prec r_{p_3}, r_{p_2} \prec r_{p_3}$
R_{12}^+	0+	0−	0−	$r_{p_2} \prec r_{p_3}$
C_2^-	0+	+−	0+	$r_{p_2} \prec r_{p_1}, r_{p_2} \prec r_{p_3}$
R_2^+	0+	0+	0+	$r_{p_2} \prec r_{p_1}, r_{p_2} \prec r_{p_3}$
R_{23}^+	0+	0+	0−	$r_{p_2} \prec r_{p_1}$
C_3^-	0+	0+	+−	$r_{p_2} \prec r_{p_1}, r_{p_3} \prec r_{p_1}, r_{p_3} \prec r_{p_2}$
R_3^+	0+	0+	0+	$r_{p_2} \prec r_{p_1}, r_{p_3} \prec r_{p_1}, r_{p_3} \prec r_{p_2}$

Table 2: Fault Signatures and Relative Measurement Orderings for the Set of Minimal Submodels of the Tank System.

Fault	$r_{p_1(p_1)}$	$r_{p_2(p_2)}$	$r_{p_3(p_3)}$	Measurement Orderings
C_1^-	+−	00	00	∅
R_1^+	0+	00	00	∅
R_{12}^+	0+	0−	00	∅
C_2^-	00	+−	00	∅
R_2^+	00	0+	00	∅
R_{23}^+	00	0+	0−	∅
C_3^-	00	00	+−	∅
R_3^+	00	00	0+	∅

Definition 4 (Fault Trace)**.** A *fault trace* for a fault f over residuals R, denoted by $\lambda_{f,R}$, is a string of length $\leq |R|$ that includes, for every $r \in R$ that will deviate due to f, a fault signature $\sigma_{f,r}$, such that the sequence of fault signatures satisfies $\Omega_{f,R}$.

We group the set of all fault traces into a *fault language*. The *fault model*, defined by a finite automaton, concisely represents the fault language of a fault.

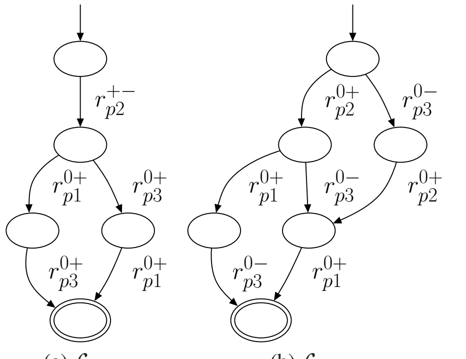
Definition 5 (Fault Language)**.** The *fault language* of a fault $f \in F$ with residual set R, denoted by $L_{f,R}$, is the set of all fault traces for f over the residuals in R.

Definition 6 (Fault Model)**.** The *fault model* for a fault $f \in F$ with residual set R, is the finite automaton that accepts exactly the language $L_{f,R}$, and is given by $\mathcal{L}_{f,R} = (S, s_0, \Sigma, \delta, A)$ where S is a set of states, $s_0 \in S$ is an initial state, Σ is a set of events, $\delta : S \times \Sigma \rightarrow S$ is a transition function, and $A \subseteq S$ is a set of accepting states.

The finite automata representation allows for the composition of the fault signatures and measurement orderings into fault models. The possible fault signatures and measurement orderings can be composed automatically to form the fault models (Daigle *et al.*, 2009). Selected fault models for a three-tank system are shown in Fig. 2. For example, as seen in $\mathcal{L}_{C_2^-,R}$, the fault C_2^- may manifest as the fault traces $r_{p_2}^{+-} r_{p_1}^{0+} r_{p_3}^{0+}$ and $r_{p_2}^{+-} r_{p_3}^{0+} r_{p_1}^{0+}$, as implied by the fault signatures and measurement orderings.

5 DISTRIBUTED DIAGNOSER DESIGN

Diagnoser design is based on the diagnosability of the system. In this work we use the notion of global diagnosability as the condition for the local diagnoser to achieve globally correct results, as in (Roychoudhury *et al.*, 2009; Daigle *et al.*, 2010). So we first define notions of diagnosability in our framework, then, we describe the diagnoser design algorithm, and finally, we summarize how we build the local event-based diagnosers.



(a) $\mathcal{L}_{C_2^-,R}$ (b) $\mathcal{L}_{R_{23}^+,R}$

Figure 2: Fault models for some faults of the three-tank system, where $R = \{r_{p_1}, r_{p_2}, r_{p_3}\}$.

5.1 Diagnosability

Given a model of a system, and the set of faults (F) and residuals (R), we may now establish the notions of *distinguishability* and *diagnosability*. Using these definitions, we can then formally define the distributed diagnoser design problem. Distinguishability between faults is characterized as follows.

Definition 7 (Distinguishability)**.** With residuals R, a fault f_i is distinguishable from a fault f_j, denoted by $f_i \not\sim_R f_j$, if f_i always eventually produces effects on the residuals that f_j cannot.

Under our framework, one fault will be distinguishable from another fault if it cannot produce a fault trace that is a prefix (denoted by $\sqsubseteq$) of a trace that can be produced by the other fault[1]. If this is not the case, then when that trace manifests, the first fault cannot be distinguished from the second.

As we previously described, the set of possible effects on residuals due to a fault is called a *fault language*. Using this definition we define a system within our framework as follows.

Definition 8 (System)**.** A *system* $\mathcal{S}$ is tuple $(F, M, R, L_{F,R})$, where $F = \{f_1, f_2, \ldots, f_n\}$ is a set of faults, M is a set of measurements, R is a set of residuals, and $L_{F,R} = \{L_{f_1,R}, L_{f_2,R}, \ldots, L_{f_n,R}\}$ is the set of fault languages.

If a system is diagnosable, then we can make guarantees about the unique isolation of every fault in the system.

[1] A fault trace λ_i is a prefix of fault trace λ_j if there is some (possibly empty) sequence of events λ_k that can extend λ_i such that $\lambda_i \lambda_k = \lambda_j$.

Definition 9 (Diagnosability)**.** A system $\mathcal{S} = (F, M, R, L_{F,R})$ is *diagnosable* if $(\forall f_i, f_j \in F) f_i \neq f_j \implies f_i \nsim_R f_j$.

If $\mathcal{S}$ is diagnosable, then every pair of faults is distinguishable using the residual set R. Hence, we can uniquely isolate all faults of interest. If $\mathcal{S}$ is not diagnosable, then ambiguities will remain after fault isolation, i.e., after all possible fault effects on the residuals have been observed. For example, consider the $\mathcal{M}$-based residual set given in Table 1. The system defined with these residuals is diagnosable when both signatures and orderings are used (without orderings, faults R_1^+, R_2^+, and R_3^+ cannot be distinguished because they all produce the same signatures). However, given the PC-based residuals (derived from the minimal submodels as shown in Table 2), the system is not diagnosable since fault R_1^+ cannot be distinguished from fault R_{12}^+, and fault R_2^+ cannot be distinguished from fault R_{23}^+. Say R_1^+ occurs, then a `0+` will be observed on $r_{p_1(p_1)}$. At this point, that observation is also consistent with R_{12}^+ occurring. No other residual will deviate in order to distinguish these two faults, so the system is not diagnosable. In this work we assume that the system is *diagnosable* for the $\mathcal{M}$-based residual set.

Our objective is to decompose the overall diagnosis task into smaller subtasks performed by local diagnosers with the following properties: *(i)* all single faults of interest in the system can be diagnosed, and *(ii)* the local diagnosis results are *globally* correct. These two properties eliminate the need for a centralized coordinator (Roychoudhury *et al.*, 2009).

The system $\mathcal{S}$ is split into n subsystems $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n$, where each fault is assigned to one subsystem, and each subsystem gets a subset of the complete measurement set and a subset of the complete residual set. Subsystem definitions are provided by the user as input.

Definition 10 (Subsystem)**.** A *subsystem* $\mathcal{S}_i$ is a tuple $(F_i, M_i, R_i, L_{F_i,R_i})$, such that *(i)* $F = F_1 \cup F_2 \cup \ldots \cup F_n$, *(ii)* $\forall i \neq j \in \{1, 2, \ldots, n\}$, $F_i \cap F_j = \varnothing$, *(iii)* $\forall i$ $M_i \subseteq M$, and *(iv)* $\forall i$ $R_i \subseteq R$.

Subsystems may be locally diagnosable. A locally diagnosable subsystem is one in which its own faults can be uniquely isolated using its own residuals. However, the local diagnosers must satisfy a notion of *global diagnosability* to achieve globally correct diagnosis (Daigle *et al.*, 2010).

Definition 11 (Global Diagnosability)**.** A subsystem $\mathcal{S}_i = (F_i, M_i, R_i, L_{F_i,R_i})$ belonging to system $\mathcal{S} = (F, M, R, L_{F,R})$ is *globally diagnosable* if $(\forall f_i \in F_i, f_j \in F) f_i \neq f_j \implies f_i \nsim_{R_i} f_j$. We say two faults $f_i \in F_i$ and $f_j \in F$ are *globally distinguishable* if $f_i \nsim_{R_i} f_j$.

That is, a subsystem $\mathcal{S}_i$ is *globally diagnosable* if all the faults F_i are distinguishable from every other fault in F using only the residuals in R_i. If the subsystems can be structured such that each subsystem $\mathcal{S}_i$ is globally diagnosable, then each local diagnoser can independently generate globally correct local diagnoses.

In this paper, we focus on the problem where $\mathcal{S}$ is already partitioned into subsystems, but each $\mathcal{S}_i$ may not be globally diagnosable. The distributed diagnoser design problem is defined as determining, for each $\mathcal{S}_i$, the minimal set of residuals to use to achieve global diagnosability. Formally, the problem is defined as:

Problem (Partitioned System Diagnoser Design)**.** Given n subsystems, where $\mathcal{S}_i = (F_i, M_i, R_i, L_{F_i,R_i})$, construct, for each subsystem, a residual set $R_i{}^+ \subseteq R$ such that *(i)* $R_i^+ - R_i$ is minimal, *(ii)* $M_i^+ \subseteq M$ are the measurements involved in R_i^+, and *(iii)* $\mathcal{S}_i' = (F_i, M_i^+, R_i^+, L_{F_i,R_i^+})$ is globally diagnosable.

5.2 Diagnoser Design Algorithm

The diagnoser design problem is, in general, a *measurement selection* problem, which is an instance of the set covering problem, known to be NP-complete (Narasimhan *et al.*, 1998). The complexity of the design problem increases with the number of residuals, and, as shown in Section 4, the complete residual set, R, grows exponentially with the number of measurements. Therefore, we need to use heuristics to guide the search.

The advantage of PCs is that they decouple the effects of all faults whose effect on the output measurement of the PC only happens through one of the measurements that are considered as input to the PC. This results in an improvement in diagnosability in a local sense. The intuition, then, is that including PC-based residuals will lead to improved diagnoser designs because of this improvement of diagnosability. So, one may simply apply the algorithm presented in (Daigle *et al.*, 2010) on the PC-based residual set. However, there are two problems. First, the system may not be diagnosable with only the PC-based residuals (see the example in the previous subsection), even if it is diagnosable with the residuals based on the global model, and second, measurement orderings cannot be derived for the PC-based residuals, so diagnostic performance may be decreased relative to a centralized diagnoser that uses measurement orderings.

Assume that the system is split into three subsystems, $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$, where for $\mathcal{S}_1$, $F_1 = \{C_1^-, R_1^+, R_{12}^+\}$, $M_1 = \{p_1\}$, for $\mathcal{S}_2$, $F_2 = \{C_2^-, R_2^+, R_{23}^+\}$, $M_2 = \{p_2\}$, and for $\mathcal{S}_3$, $F_3 = \{C_3^-, R_3^+\}$, $M_3 = \{p_3\}$. Assume we use the $\mathcal{M}$-based residuals, so $R_1 = \{r_{p_1}\}$, $R_2 = \{r_{p_2}\}$, and $R_3 = \{r_{p_3}\}$. Analyzing global diagnosability, we see that none of the subsystems is globally diagnosable, i.e., we will have to add new residuals to each subsystem in order to satisfy our design constraints. For example, consider subsystem $\mathcal{S}_3$. Looking at Table 1, we see that the subsystem is not globally diagnosable, since effects produced by R_3^+ on r_{p_3} are not unique.

Now assume we use the PC-based residuals, $R_1 = \{r_{p_1(p_1)}\}$, $R_2 = \{r_{p_2(p_2)}\}$, and $R_3 = \{r_{p_3(p_3)}\}$. We see that now $\mathcal{S}_3$ is globally diagnosable because only one nonlocal fault, R_{23}^+, produces an effect on $r_{p_3(p_3)}$, and it is a different effect from those produced by the local faults. So if $\mathcal{S}_3$ uses the PC-based residual instead of the global model-based residual, it can have an improved diagnoser design. But, the other subsystems are not globally diagnosable, and cannot be made so by including any other PC-based residual, because those subsystems contain the faults that make the system as a whole nondiagnosable using only the PC-based resid-

Algorithm 1 Distributed Diagnoser Design

```
Input: $\mathbb{S} = \{\mathcal{S}_i = (F_i, M_i, \varnothing, \varnothing) : i = 1, \ldots, n\}$
for all $\mathcal{S}_i \in \mathbb{S}$ do
    $R_i \leftarrow \{r_{m(M_i)} : m \in M_i\}$
    while $\mathcal{S}_i$ not globally diagnosable do
        $M' \leftarrow \texttt{computeMSubset}(M_i, M)$
        $M_i^* \leftarrow \texttt{findBestM}(F, F_i, M', M_i)$
        $M_i \leftarrow M_i \cup M_i^*$
        $R_i \leftarrow \{r_{m(M_i^+)} : m \in M_i^+\}$
    end while
    construct $\mathcal{D}_{F_i,R_i}$
end for
```

uals. This suggests that we require a more general approach that combines both PC-based residuals and the $\mathcal{M}$-based residuals. In general, we need to consider residuals from the complete set considering all possible submodels.

But, as previously pointed out, the complexity of the design problem is dependent on the number of possible residuals, and the complete set is too large. Further, there is much overlap of information between the different residuals, for instance, compare Tables 1 and 2. Instead, we perform a search over the *measurement* space, which is much smaller, and define residuals in a particular way for a given set of measurements. Specifically, given a set of measurements M_i, we use the residuals for the submodel that includes exactly the measurements in M_i as outputs, i.e., for M_i we use the residual set $\{r_{m(M_i)} : m \in M_i\}$. We then incrementally expand the submodel of each subsystem to include additional measurements (and, hence, a larger set of residuals) in order to satisfy global diagnosability.

The diagnoser design algorithm is shown in Algorithm 1. For each subsystem, we first construct the set of residuals for its current measurement set. We then determine a subset of measurements over $M' \subseteq M - M_i$ over which we will consider adding to the subsystem using the `computeMSubset` function. In our particular implementation, we simply set M' equal to $M - M_i$, but, in general, this may include heuristics such as the subsystem distance heuristic developed in (Roychoudhury *et al.*, 2009). We then identify the best (with respect to global diagnosability) subset of measurements M_i^* within M' to add to M_i, using the `findBestM` function. We then update M_i, reconstruct the residual set for the new measurement set, and continue in this fashion until $\mathcal{S}_i$ is globally diagnosable[2]. Then, the algorithm constructs $\mathcal{D}_{F_i,R_i}$ to build the local diagnoser. This procedure is explained in the next subsection.

In our particular implementation, we used the `findBestM` function shown as Function 2. Here, we select only the single best measurement, rather than a subset of measurements. For each possible measurement to add, we construct the new set of residuals, then determine the set of faults F_i^* that are not globally distinguishable for the subsystem and this residual set. The measurement that results in the smallest F_i^* is selected as the best measurement and becomes the output M_i^*. Adding measurements incrementally, and especially one at a time, is, in general, nonoptimal, but here we trade off optimality for computational efficiency. More complex versions of this function are also possible.

Function 2 $M_i^* \leftarrow \texttt{findBestM}(F, F_i, M, M_i)$

```
for all $m \in M - M_i$ do
    $R_i \leftarrow \{r_{m'(M_i \cup \{m\})} : m' \in M_i \cup \{m\}\}$
    $F_i^* \leftarrow \{f_i^* : f_i \sim_{R_i} f_j$ for $f_i^* \in F_i, f_j \in F$, and $f_i^* \neq f_j\}$
    $score_m \leftarrow |F_i^*|$
end for
$M_i^* \leftarrow \{m : score_m$ is minimum$\}$
```

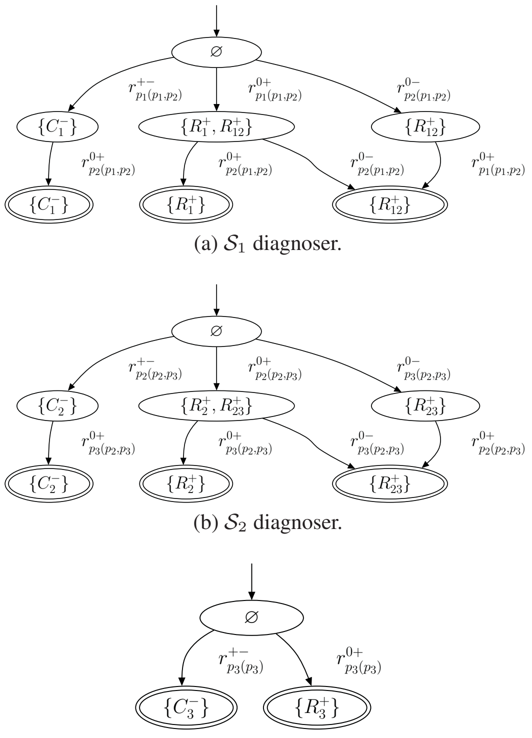
We apply this algorithm to the three-tank system, where for tank i, for $i = 1, \ldots, n-1$, $\mathcal{S}_i$ is defined by $F_i = \{C_i^-, R_i^+, R_{i,i+1}^+\}$ and $M_i = \{p_i\}$, and for $i = n$, $\mathcal{S}_i$ is defined by $F_i = \{C_i^-, R_i^+\}$ and $M_i = \{p_i\}$. As a result, we have to add one residual only to the subsystems $\mathcal{S}_1$ and $\mathcal{S}_2$, and none have to be added to subsystem $\mathcal{S}_3$, because, as shown previously, the subsystem is already globally diagnosable with only $r_{p_3(p_3)}$. Subsystem $\mathcal{S}_1$ gets residuals $r_{p_1(p_1,p_2)}$ and $r_{p_2(p_1,p_2)}$, and subsystem $\mathcal{S}_2$ gets residuals $r_{p_2(p_2,p_3)}$ and $r_{p_3(p_2,p_3)}$. This improves the algorithm presented in (Daigle *et al.*, 2010), because in that case, subsystem $\mathcal{S}_2$ needs three residuals, and subsystem $\mathcal{S}_3$ needs two residuals, so the size of the event-based diagnosers is improved.

There is also a second way in which the design is improved over the approach of (Daigle *et al.*, 2010). In that approach, each subsystem uses the global model for residual generation. In the approach developed in this paper, however, each subsystem needs only a submodel for residual generation. So, residual generation will be more efficient. In fact, this will always be the case, because the only time a subsystem will end up using the global model is if it adds all the measurements to the subsystem. This is a worst-case design, and, on average, each subsystem will only use a subset of the measurements, and, therefore, a subset of the global model for residual generation.

5.3 Diagnoser Implementation

Once we have designed the distributed diagnosis system, event-based diagnosers may be constructed. An event-based diagnoser, $\mathcal{D}_{F,R}$, for fault set F and residual set R, is a finite automaton extended by a set of diagnoses and a diagnosis map and is similar in concept to DES diagnosers such as in (Sampath *et al.*, 1996). It takes events as inputs, which, as with fault models, correspond to residual deviations. From the current state, a residual deviation event causes a transition to a new state. The diagnosis for that new state represents the set of faults that are consistent with the sequence of events seen up to the current point in time. The diagnoser is constructed to capture the fault languages and link fault traces to diagnoses. Details of this procedure can be found in (Daigle *et al.*, 2009). The design of local diagnosers follows the same procedure as the global diagnoser, i.e., given F_i and R_i for subsystem $\mathcal{S}_i$, we construct $\mathcal{D}_{F_i,R_i}$. The local diagnosers for the distributed diagnoser design example for the three-tank system are given in Fig. 3. Accepting states correspond to globally correct diagnosis.

[2] The algorithm will always end successfully, since, in the worst case, the global model will be regained, which we have assumed to be globally diagnosable.



(a) $\mathcal{S}_1$ diagnoser.

(b) $\mathcal{S}_2$ diagnoser.

(c) $\mathcal{S}_3$ diagnoser.

Figure 3: Local diagnosers for the three-tank system for $F_1 = \{C_1^-, R_1^+, R_{12}^+\}$, $R_1 = \{r_{p_1(p_1,p_2)}, r_{p_2(p_1,p_2)}\}$, $F_2 = \{C_2^-, R_2^+, R_{23}^+\}$, $R_2 = \{r_{p_2(p_2,p_3)}, r_{p_3(p_2,p_3)}\}$, $F_3 = \{C_3^-, R_3^+\}$ and $R_3 = \{r_{p_3(p_3)}\}$.

6 RESULTS

This section shows the applicability of the proposed design approach. First, we show different design scenarios and compare the design obtained with the new approach against the design obtained using the previous approach in (Daigle *et al.*, 2010). Then we show an example to demonstrate online diagnosis.

6.1 Distributed Diagnoser Design Experiments

As a first design scenario, consider the three-tank system with $F = \{C_1^-,C_2^-,C_3^-,R_{12}^+,R_{23}^+\}$ and $M = \{p_1,p_2,p_3\}$. Now, assume that the system is split into three subsystems, $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$, where for $\mathcal{S}_1$, $F_1 = \{C_1^-,R_{12}^+\}$, $M_1 = \{p_1\}$, for $\mathcal{S}_2$, $F_2 = \{C_2^-, R_{23}^+\}$, $M_2 = \{p_2\}$, and for $\mathcal{S}_3$, $F_3 = \{C_3^-\}$, $M_3 = \{p_3\}$. If we use the PC-based residuals, $R_1 = \{r_{p_1(p_1)}\}$, $R_2 = \{r_{p_2(p_2)}\}$, and $R_3 = \{r_{p_3(p_3)}\}$ we see that all three subsystems, $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$, are globally diagnosable. This is clear from the set of fault signatures obtained using these residuals, shown in Table 2. The PCs decouple the subsystems to the extent that only the R_{ij}^+ faults affect multiple subsystems, and the effects they produce are unique. Hence, no design is needed in this case, and we will be able to use the minimal PC-based residuals instead of the global model-based residuals. This improves over the previous approach, because in that case, subsystem $\mathcal{S}_1$ needs two residuals, and subsystem $\mathcal{S}_2$ also needs two residuals, so the size of the event-based diagnosers is improved and the search process is completely avoided.

On the other hand, consider that we have now a new scenario with $F = \{C_1^-, C_1^+, C_2^-, C_2^+, C_3^-, C_3^+, R_1^-, R_1^+, R_2^-, R_2^+, R_3^-, R_3^+, R_{12}^-, R_{12}^+, R_{23}^-, R_{23}^+\}$, $M = \{p_1,q_2,q_3\}$, and $R = \{r_{p_1}, r_{q_2}, r_{q_3}\}$. Assume that the system is split into three subsystems, $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$, where for $\mathcal{S}_1$, $F_1 = \{C_1^-, C_1^+, R_1^-, R_1^+, R_{12}^-, R_{12}^+\}$, $M_1 = \{p_1\}$, for $\mathcal{S}_2$, $F_2 = \{C_2^-,C_2^+,R_2^-,R_2^+,R_{23}^-,R_{23}^+\}$, $M_2 = \{q_2\}$, and for $\mathcal{S}_3$, $F_3 = \{C_3^-,C_3^+,R_3^-,R_3^+\}$, $M_3 = \{q_3\}$. If we use the PC-based residuals, $R_1 = \{r_{p_1(p_1)}\}$, $R_2 = \{r_{q_2(q_2)}\}$, and $R_3 = \{r_{q_3(q_3)}\}$ none of the subsystems is globally diagnosable, and we have to apply our new design algorithm, that results in adding one residual to each subsystem. Subsystem $\mathcal{S}_1$ gets residuals $r_{p_1(p_1,q_2)}$ and $r_{q_2(p_1,q_2)}$, subsystem $\mathcal{S}_2$ gets residuals $r_{q_2(q_2,q_3)}$ and $r_{q_3(q_2,q_3)}$, and subsystem $\mathcal{S}_3$ gets residuals $r_{q_3(q_2,q_3)}$ and $r_{q_2(q_2,q_3)}$. The diagnoser size is the same as with the algorithm presented in (Daigle *et al.*, 2010), but the new approach is still an improvement because the local residual generation process is more efficient, since each subsystem uses only a submodel.

We ran additional experiments with different design criteria, and we found that the size of the local diagnosers with the new approach was always less than or equal to the size of the local diagnosers obtained using the approach in (Daigle *et al.*, 2010).

6.2 On-line Fault Diagnosis

As an example to demonstrate online diagnosis in this framework, consider the three-tank system example from Section 5, with R_2^+ occurring at time 10.0 seconds. Fig. 4 shows the plots of the residuals that are triggered by this fault ($r_{p_1(p_1,p_2)}$, $r_{p_2(p_1,p_2)}$, $r_{p_2(p_2,p_3)}$ and $r_{p_3(p_2,p_3)}$). At time 10.2 s, an increase in residual $r_{p_2(p_1,p_2)}$ is detected in $\mathcal{S}_1$ and in $r_{p_2(p_2,p_3)}$ by $\mathcal{S}_2$ (Fig. 3 shows the local diagnosers). The $\mathcal{S}_1$ diagnoser blocks on the first state, i.e., it eliminates all fault candidates, since the only possible deviation considered in residual $r_{p_2(p_1,p_2)}$ by the local diagnoser is a $-$. For $\mathcal{S}_2$, the local diagnoser simultaneously moves to the state with diagnosis $\{C_2^-\}$, and the state with diagnosis $\{R_2^+, R_{23}^+\}$ since the full signature is not yet known. At 10.6 s, an increase in $r_{p_3(p_2,p_3)}$ is detected and the diagnoser moves to the states with diagnosis $\{C_2^-\}$ and $\{R_2^+\}$. At time 11.2 s it is determined that the initial change in $r_{p_2(p_2,p_3)}$ was smooth, resulting in a signature of $0+$. Hence, the hypothesized path to the state with $\{C_2^-\}$ is eliminated and the diagnosis is confirmed as $\{R_2^+\}$. Since the diagnoser has reached to an accepting state, a global diagnosis has been achieved.

7 CONCLUSIONS

In this work we developed a new framework for distributed event-based qualitative diagnosis of continuous systems using structural model decomposition. PCs are

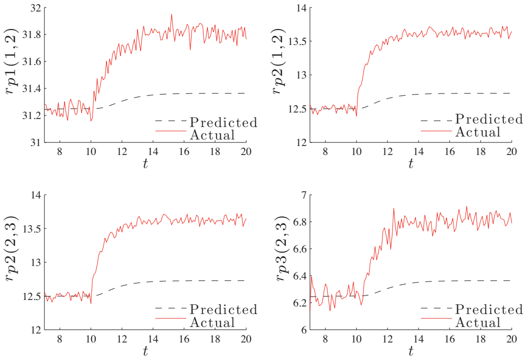


Figure 4: Three-tank predicted and observed flow outputs $r_{p_1(p_1,p_2)}$ and $r_{p_2(p_1,p_2)}$ for $\mathcal{S}_1$, and $r_{p_2(p_2,p_3)}$ and $r_{p_3(p_2,p_3)}$ for $\mathcal{S}_2$.

used to decouple the system and compute minimal submodels for diagnosis. Then, the basic PC framework is extended to allow PC merging to design globally diagnosable subsystems. We proposed an algorithm that merges minimal submodels (when necessary) to design the distributed diagnosers based on global diagnosability. The approach builds on that presented in (Daigle *et al.*, 2010), so results also in a distributed diagnosis framework that has no single point of failure and scales well. Experimental results on a multi-tank system show the improvement of the design using submodels against the previous approach using the global model of the system (Daigle *et al.*, 2010). Experiments show a decrease in the size of the event-based diagnosers. Moreover, since the proposed approach uses submodels, the residual generation process is more efficient and the residual generators for subsystems are fully decoupled.

The distributed diagnosis framework relates to distributed discrete-event system (DES) diagnosis methods like (Debouk *et al.*, 2000). The local diagnosers are designed to provide globally correct diagnosis results, contrasting with other DES approaches such as (Pencolé and Cordier, 2005), where a merge operation of diagnosis results is necessary to obtain the global diagnosis. The abstraction of the continuous dynamics into an event-based representation is also similar to (Meseguer *et al.*, 2010; Bayoudh *et al.*, 2006).

In future work, we will integrate the proposed approach within a diagnosis framework that goes from fault detection to fault identification, where we will exploit additional properties of the minimal submodels (like the computation of minimal parameter estimators for fault identification (Bregon *et al.*, to appear)).

ACKNOWLEDGMENTS

Anibal Bregon and Belarmino Pulido's work has been funded by the Spanish MCI DPI2008-01996 and TIN2009-11326 grants.

REFERENCES

(Bayoudh *et al.*, 2006) M. Bayoudh, L. Travé-Massuyès, and X. Olive. Hybrid systems diagnosability by abstracting faulty continuous dynamics. In *Proc. of the 17th Int. Workshop on Principles of Diagnosis*, pages 9–15, 2006.

(Bregon *et al.*, to appear) A. Bregon, G. Biswas, and B. Pulido. A Decomposition Method for Nonlinear Parameter Estimation in TRANSCEND. *IEEE Trans. Syst. Man Cy. A.*, to appear.

(Daigle *et al.*, 2009) M. J. Daigle, X. Koutsoukos, and G. Biswas. A qualitative event-based approach to continuous systems diagnosis. *IEEE Trans. Control Syst. Technol.*, 17(4):780–793, July 2009.

(Daigle *et al.*, 2010) M. Daigle, I. Roychoudhury, G. Biswas, and X. Koutsoukos. An Event-based Approach to Distributed Diagnosis of Continuous Systems. In *Proceedings of the 21th International Workshop on Principles of Diagnosis, DX10*, pages 15–22, Portland, Oregon, USA, 2010.

(Daigle, 2008) M. Daigle. *A Qualitative Event-based Approach to Fault Diagnosis of Hybrid Systems*. PhD thesis, Vanderbilt University, 2008.

(Debouk *et al.*, 2000) Rami Debouk, S. Lafortune, and Demosthenis Teneketzis. Coordinated decentralized protocols for failure diagnosis of discrete event systems. *Discrete Event Dynamic Systems*, 10(1-2):33–86, 2000.

(Meseguer *et al.*, 2010) J. Meseguer, V. Puig, and T. Escobet. Fault Diagnosis Using a Timed Discrete-Event Approach Based on Interval Observers: Application to Sewer Networks. *IEEE Trans. Syst. Man Cy. A.*, 40(5):900–916, 2010.

(Mosterman and Biswas, 1999) P. J. Mosterman and G. Biswas. Diagnosis of continuous valued systems in transient operating regions. *IEEE Trans. Syst. Man Cy. A.*, 29(6):554–565, 1999.

(Narasimhan *et al.*, 1998) S. Narasimhan, P. J. Mosterman, and G. Biswas. A systematic analysis of measurement selection algorithms for fault isolation in dynamic systems. In *Proc. of DX 1998*, pages 94–101, Cape Cod, MA USA, May 1998.

(Pencolé and Cordier, 2005) Yannick Pencolé and Marie-Odile Cordier. A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks. *Artif. Intell.*, 164:121–170, May 2005.

(Pulido and Alonso-González, 2004) B. Pulido and C. Alonso-González. Possible Conflicts: a compilation technique for consistency-based diagnosis. *IEEE Trans. Syst. Man Cy. B.*, 34(5):2192–2206, 2004.

(Roychoudhury *et al.*, 2009) I. Roychoudhury, G. Biswas, and X. Koutsoukos. Designing distributed diagnosers for complex continuous systems. *IEEE Trans. Autom. Sci. Eng.*, 6(2):277–290, April 2009.

(Sampath *et al.*, 1996) M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D.C. Teneketzis. Failure diagnosis using discrete-event models. *IEEE Trans. Control Syst. Technol.*, 4(2):105–124, Mar. 1996.